**이름 –** 이상호

**학번 –** 12212312

**학과 –** 경영학과

**데이터 –** [Default of Credit Card Clients Dataset | Kaggle](https://www.kaggle.com/datasets/uciml/default-of-credit-card-clients-dataset)

**분석 목표 –** 연령, 성별, 교육 수준, 결혼 여부에 따라 신용카드 사용액, 상환 여부, 한도 등이 어떻게 달라지는 지를 분석하고자 한다.

**분석 내용 –** 이 데이터는 대만의 어떤 신용카드사에서 고객들의 신용카드 사용 정보, 신상 정보들을 모아놓은 것으로 2005년 4월부터 9월까지의 데이터가 존재한다. 이 데이터는 다음과 같은 변수로 이루어져 있다.

ID: ID of each client

LIMIT\_BAL: Amount of given credit in NT dollars (includes individual and family/supplementary credit)

SEX: Gender (1=male, 2=female)

EDUCATION: (1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)

MARRIAGE: Marital status (1=married, 2=single, 3=others)

AGE: Age in years

PAY\_0: Repayment status in September 2005 (-1=pay duly, 1=payment delay for one month, 2=payment delay for two months, … 8=payment delay for eight months, 9=payment delay for nine months and above)

PAY\_2: Repayment status in August 2005 (scale same as above)

PAY\_3: Repayment status in July 2005 (scale same as above)

PAY\_4: Repayment status in June 2005 (scale same as above)

PAY\_5: Repayment status in May 2005 (scale same as above)

PAY\_6: Repayment status in April 2005 (scale same as above)

BILL\_AMT1: Amount of bill statement in September 2005 (NT dollar)

BILL\_AMT2: Amount of bill statement in August 2005 (NT dollar)

BILL\_AMT3: Amount of bill statement in July 2005 (NT dollar)

BILL\_AMT4: Amount of bill statement in June 2005 (NT dollar)

BILL\_AMT5: Amount of bill statement in May 2005 (NT dollar)

BILL\_AMT6: Amount of bill statement in April 2005 (NT dollar)

PAY\_AMT1: Amount of previous payment in September 2005 (NT dollar)

PAY\_AMT2: Amount of previous payment in August 2005 (NT dollar)

PAY\_AMT3: Amount of previous payment in July 2005 (NT dollar)

PAY\_AMT4: Amount of previous payment in June 2005 (NT dollar)

PAY\_AMT5: Amount of previous payment in May 2005 (NT dollar)

PAY\_AMT6: Amount of previous payment in April 2005 (NT dollar)

default.payment.next.month: Default payment (1=yes, 0=no)

우선 모든 변수에 값이 존재하므로 결측치를 채울 필요는 없다.

print(dataframe.isnull().sum())

>>> LIMIT\_BAL 0

SEX 0

EDUCATION 0

MARRIAGE 0

AGE 0

PAY\_0 0

PAY\_2 0

PAY\_3 0

PAY\_4 0

PAY\_5 0

PAY\_6 0

BILL\_AMT1 0

BILL\_AMT2 0

BILL\_AMT3 0

BILL\_AMT4 0

BILL\_AMT5 0

BILL\_AMT6 0

PAY\_AMT1 0

PAY\_AMT2 0

PAY\_AMT3 0

PAY\_AMT4 0

PAY\_AMT5 0

PAY\_AMT6 0

default.payment.next.month 0

그리고 시각화를 위해서는 변수 이름을 알아보기 쉽게 만드는 것이 좋다. 우선 해당 월의 카드 대금 상환 여부를 나타내는 PAY\_ 변수의 이름을 REPAYMENT STATUS\_ 로 바꾼다.

dataframe.rename(columns = {'PAY\_0' : 'REPAYMENT STATUS\_SEP', 'PAY\_2' : 'REPAYMENT STATUS\_AUG', 'PAY\_3' : 'REPAYMENT STATUS\_JUL'  
 , 'PAY\_4' : 'REPAYMENT STATUS\_JUN', 'PAY\_5' : 'REPAYMENT STATUS\_MAY', 'PAY\_6' : 'REPAYMENT STATUS\_APR'}, inplace = True)

해당 월의 카드 대금 청구액을 나타내는 BILL\_AMT 변수의 이름은 CARD BILL\_ 로 바꾼다.

dataframe.rename(columns = {'BILL\_AMT1' : 'CARD BILL\_SEP', 'BILL\_AMT2' : 'CARD BILL\_AUG', 'BILL\_AMT3' : 'CARD BILL\_JUL'  
 , 'BILL\_AMT4' : 'CARD BILL\_JUN', 'BILL\_AMT5' : 'CARD BILL\_MAY', 'BILL\_AMT6' : 'CARD BILL\_APR'}, inplace = True)

해당 월의 선불결제 이용 금액을 나타내는 PAY\_AMT 변수의 이름을 PREPAID\_ 로 바꾼다.

dataframe.rename(columns = {'PAY\_AMT1' : 'PREPAID\_SEP', 'PAY\_AMT2' : 'PREPAID\_AUG', 'PAY\_AMT3' : 'PREPAID\_JUL'  
 , 'PAY\_AMT4' : 'PREPAID\_JUN', 'PAY\_AMT5' : 'PREPAID\_MAY', 'PAY\_AMT6' : 'PREPAID\_APR'}, inplace = True)

마지막으로 고객의 신용한도를 나타내는 LIMIT\_BAL변수의 이름을 CREDIT LIMIT으로, 고객의 현재 체납 여부를 나타내는 default.payment.next.month 변수의 이름을 DEFAULT 로 바꾼다.

dataframe.rename(columns = {'LIMIT\_BAL' : 'CREDIT LIMIT', 'default.payment.next.month' : 'DEFAULT'}, inplace = True)

또한 SEX, EDUCATION, MARRIAGE, REPAYMENT STATUS\_ 등은 숫자로 표시되어있지만 사실은 범주 변수이므로 그에 맞게 관측값의 이름도 바꾼다.

dataframe.replace({'SEX' : 1}, 'Male', inplace = True)  
dataframe.replace({'SEX' : 2}, 'Female', inplace = True)  
dataframe.replace({'EDUCATION' : 1}, 'Graduate', inplace = True)  
dataframe.replace({'EDUCATION' : 2}, 'Undergraduate', inplace = True)  
dataframe.replace({'EDUCATION' : 3}, 'High School', inplace = True)  
dataframe.replace({'EDUCATION' : 4}, 'Less than Middle School', inplace = True)  
dataframe.replace({'EDUCATION' : 5}, 'Unknown', inplace = True)  
dataframe.replace({'EDUCATION' : 6}, 'Unknown', inplace = True)  
dataframe.replace({'MARRIAGE' : 1}, 'Married', inplace = True)  
dataframe.replace({'MARRIAGE' : 2}, 'Single', inplace = True)  
dataframe.replace({'MARRIAGE' : 3}, 'Others', inplace = True)  
RepayStatVariable = ['REPAYMENT STATUS\_SEP', 'REPAYMENT STATUS\_AUG', 'REPAYMENT STATUS\_JUL', 'REPAYMENT STATUS\_JUN', 'REPAYMENT STATUS\_MAY', 'REPAYMENT STATUS\_APR']  
for i in RepayStatVariable:  
 dataframe.replace({i : -1}, 'No Overdue', inplace=True)  
 dataframe.replace({i : 1}, 'Short-term Overdue', inplace = True)  
 dataframe.replace({i : 2}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 3}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 4}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 5}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 6}, 'Long-term Overdue', inplace = True)  
 dataframe.replace({i : 7}, 'Long-term Overdue', inplace=True)  
 dataframe.replace({i : 8}, 'Long-term Overdue', inplace=True)  
 dataframe.replace({i : 9}, 'Long-term Overdue', inplace=True)

dataframe.replace({'DEFAULT': 0}, 'Yes', inplace=True)  
dataframe.replace({'DEFAULT': 1}, 'No', inplace=True)

학력을 알 수 없는 고객과 결혼상태가 Others인 고객은 분석에 적합하지 않으므로 삭제한다.

또 관측값이 0으로 되어있는 경우에는 사실상 결측치이므로 삭제한다.

UnkownCustomer = dataframe.loc[(dataframe['EDUCATION'] == 'Unknown') | (dataframe['EDUCATION'] == 0) | (dataframe['MARRIAGE'] == 'Others') | (dataframe['MARRIAGE'] == 0)].index  
dataframe.drop(UnkownCustomer, inplace = True)

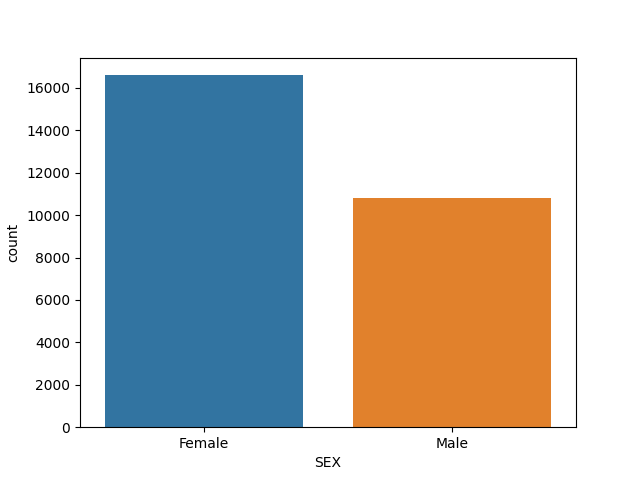
REPAYMENT STATUS\_ 에서 0과 -2는 데이터 범례에서 확인할 수 없었다. 보통 연체를 하지 않는 이용자가 대부분인 경우가 많은 점, 0개월 연체나 -2개월 연체는 불가능 하다는 점을 고려해 둘다 모두 연체가 없다고 기록한다.

RepayStatVariable = ['REPAYMENT STATUS\_SEP', 'REPAYMENT STATUS\_AUG', 'REPAYMENT STATUS\_JUL', 'REPAYMENT STATUS\_JUN', 'REPAYMENT STATUS\_MAY', 'REPAYMENT STATUS\_APR']  
for i in RepayStatVariable:  
 dataframe.replace({i : 0}, 'No Overdue', inplace=True)  
 dataframe.replace({i : -2}, 'No Overdue', inplace = True)

CARD BILL\_ 이 0 미만인 경우는 상식적으로 설명이 불가능하므로 해당 고객은 삭제한다.

MinersBillCustomer = dataframe.loc[(dataframe['CARD BILL\_SEP'] < 0) | (dataframe['CARD BILL\_AUG'] < 0) | (dataframe['CARD BILL\_JUL'] < 0) |  
 (dataframe['CARD BILL\_JUN'] < 0) | (dataframe['CARD BILL\_MAY'] < 0) | (dataframe['CARD BILL\_APR'] < 0)].index  
dataframe.drop(MinersBillCustomer, inplace = True)

이상으로 데이터 전처리는 끝났다. 이제 어떤 신용카드사에서 고객의 인구통계학적 정보에 따라 신용결제액, 선불결제액, 대금 상환 가능성, 채무불이행 가능성 등이 어떻게 다른지를 분석해본 다고 가정하고 데이터를 시각화 해보자

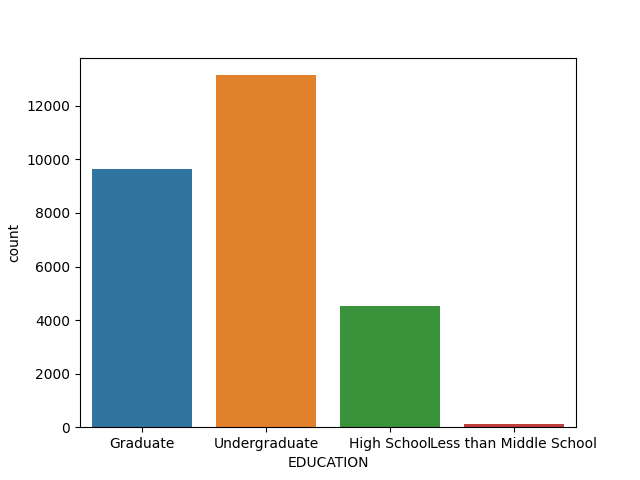
일단 분석에 앞서 인규통계학적 변수를 범주로하여 표본의 특성을 살펴보면,

print(dataframe['SEX'].value\_counts())

>>> Female 16599

Male 10798

표본에서 여성은 61%, 남성은 39%로 여성이 더 많았다.

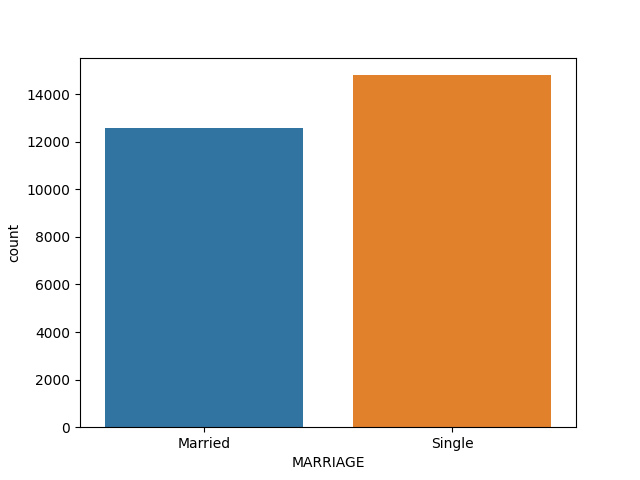
print(dataframe['EDUCATION'].value\_counts())

>>> Graduate 9638

Undergraduate 13144

High School 4508

Less than Middle School 107

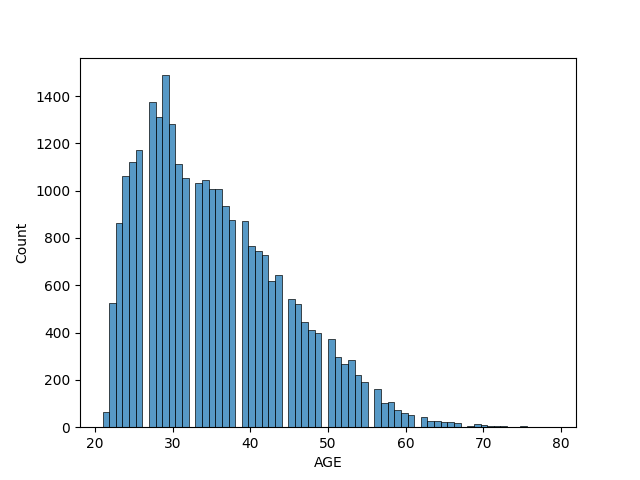
표본에서 학력은 대학생이 약 50%로 제일 많았으며, 대졸자가 약 36%로 그 다음이었고, 고졸자는 약 15% 가량이었으며, 중졸 이하는 거의 없었다.

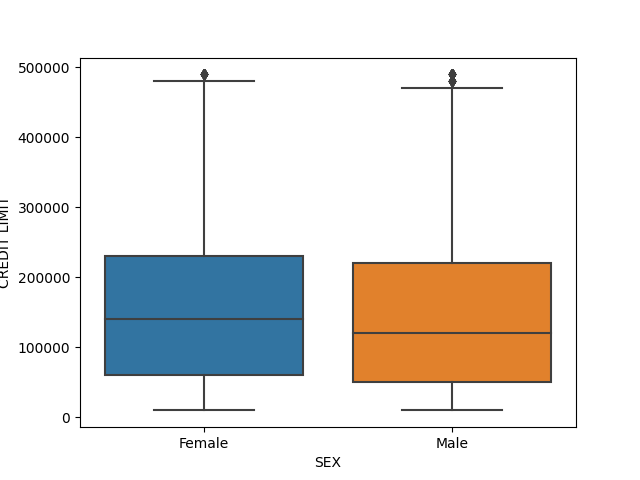
print(dataframe['EDUCATION'].value\_counts())

>>> Single 14798

Married 12599

표본에서 결혼여부는 기혼 56%, 미혼 44%로 큰차이는 없었다.

표본에서 연령은 20대 ~ 40대가 가장 많았고 그중 최빈값은 20대 후반 ~ 30대 초반 이었으며 그 이후로 점차 줄어드는 오른쪽 꼬리 분포를 보였다.

우선 인구통계학적 변수 중 성별을 범주로 하여 신용카드 사용 패턴을 알아보자

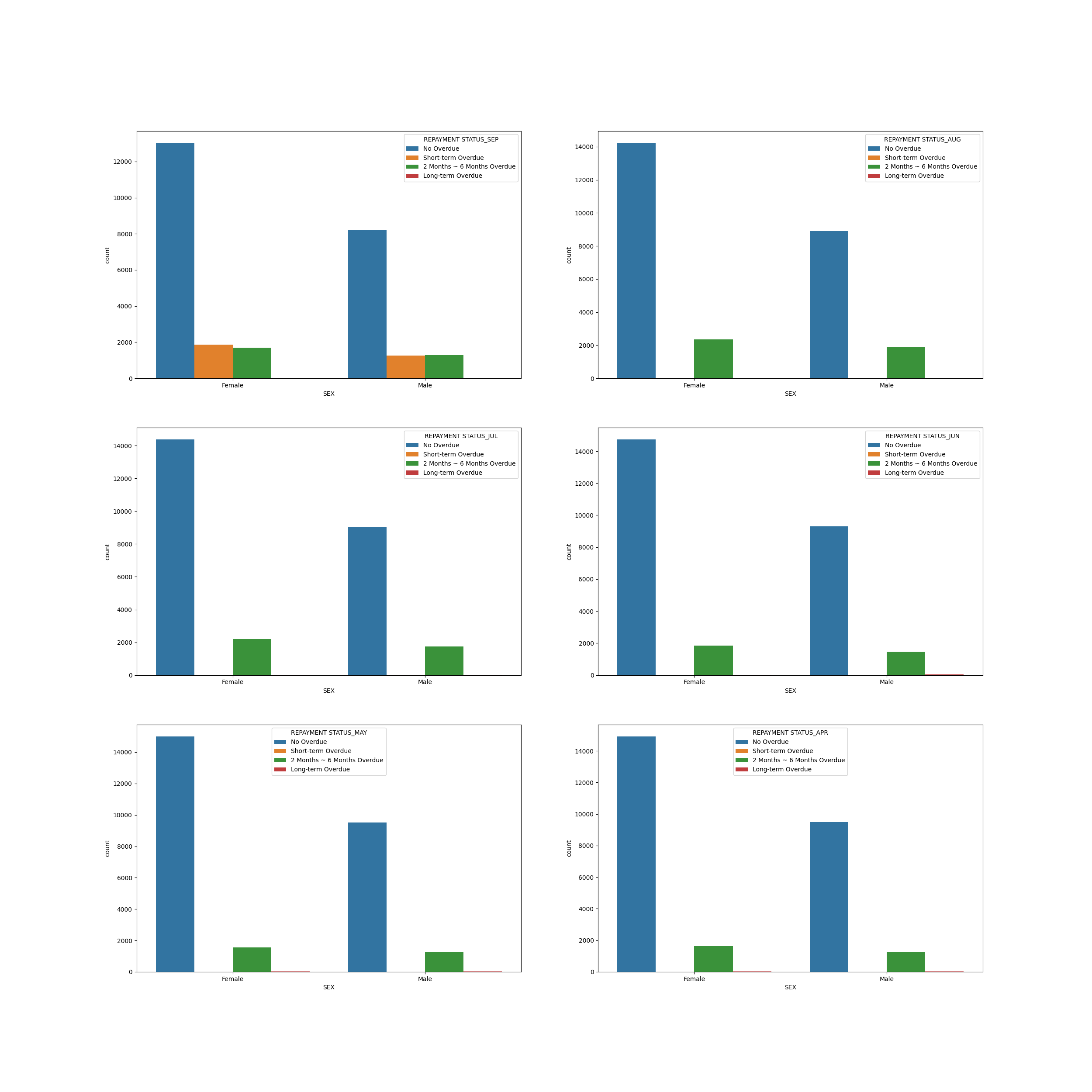
Q1\_s1 = dataframe['CREDIT LIMIT'].quantile(0.25)  
Q3\_s1 = dataframe['CREDIT LIMIT'].quantile(0.75)  
IQR\_s1 = Q3\_s1 - Q1\_s1  
sns.boxplot(x = dataframe['SEX'], y = dataframe.loc[(dataframe['CREDIT LIMIT'] > Q1\_s1 - 1.5 \* IQR\_s1)  
 & (dataframe['CREDIT LIMIT'] < Q3\_s1 + 1.5 \* IQR\_s1)]['CREDIT LIMIT'], data = dataframe)  
plt.show()  
print(dataframe.groupby(['SEX']).median())

>>> CREDIT LIMIT

SEX

Female 140000.0

Male 120000.0

아무 처리를 하지 않은 신용한도 데이터에는 이상치가 매우 많았다. 따라서 해당 이상 치를 제거한 후 시각화를 진행했다. 우선 성별에 따른 신용한도의 차이는 크지 않았다. 여성의 신 용한도가 중앙값 기준으로 약 16.6% 더 높았으나 크게 의미있는 차이라고 보기 힘들다. 평균 적으로 여성이 약간 더 높은 한도를 받는다는 정도로만 볼 수 있다.

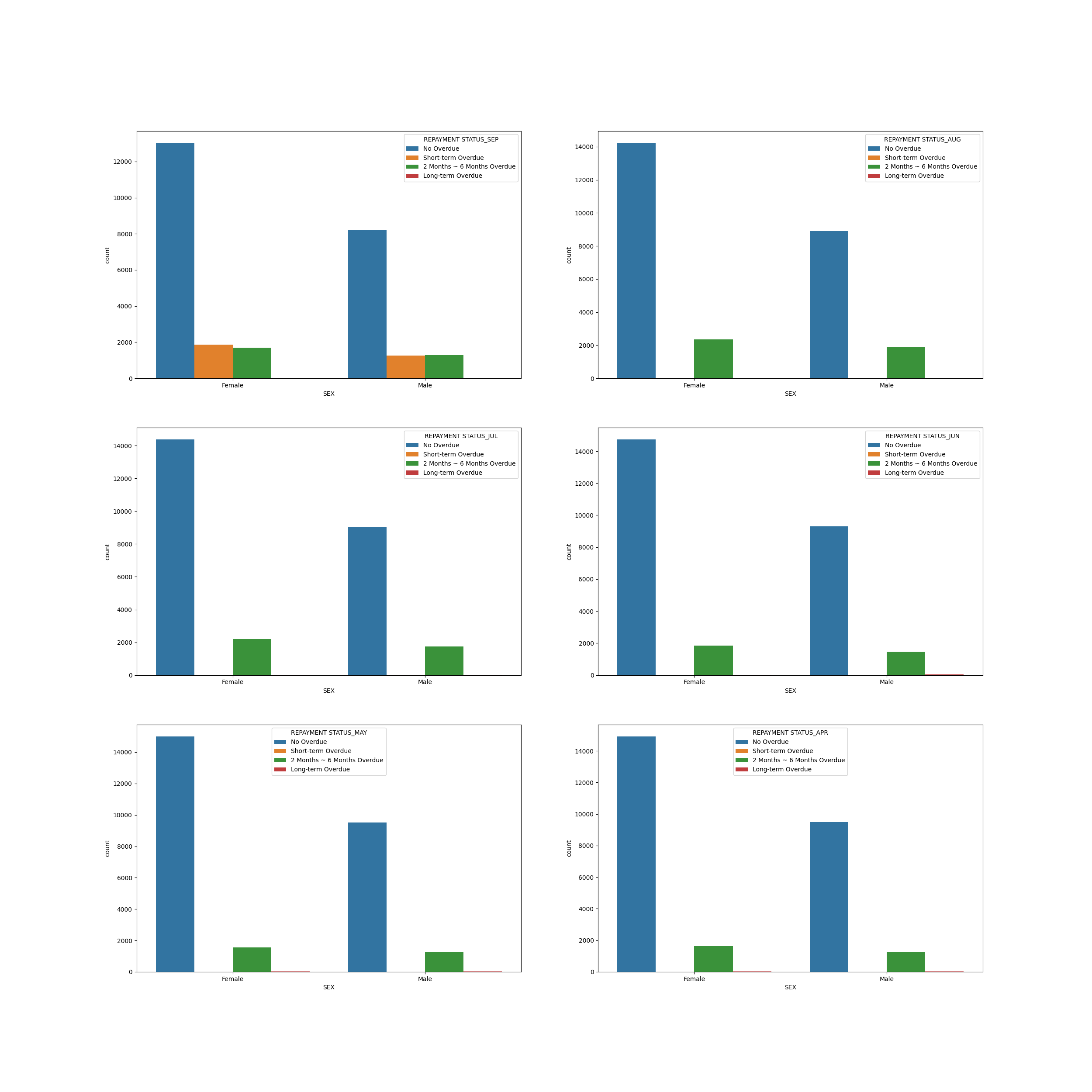
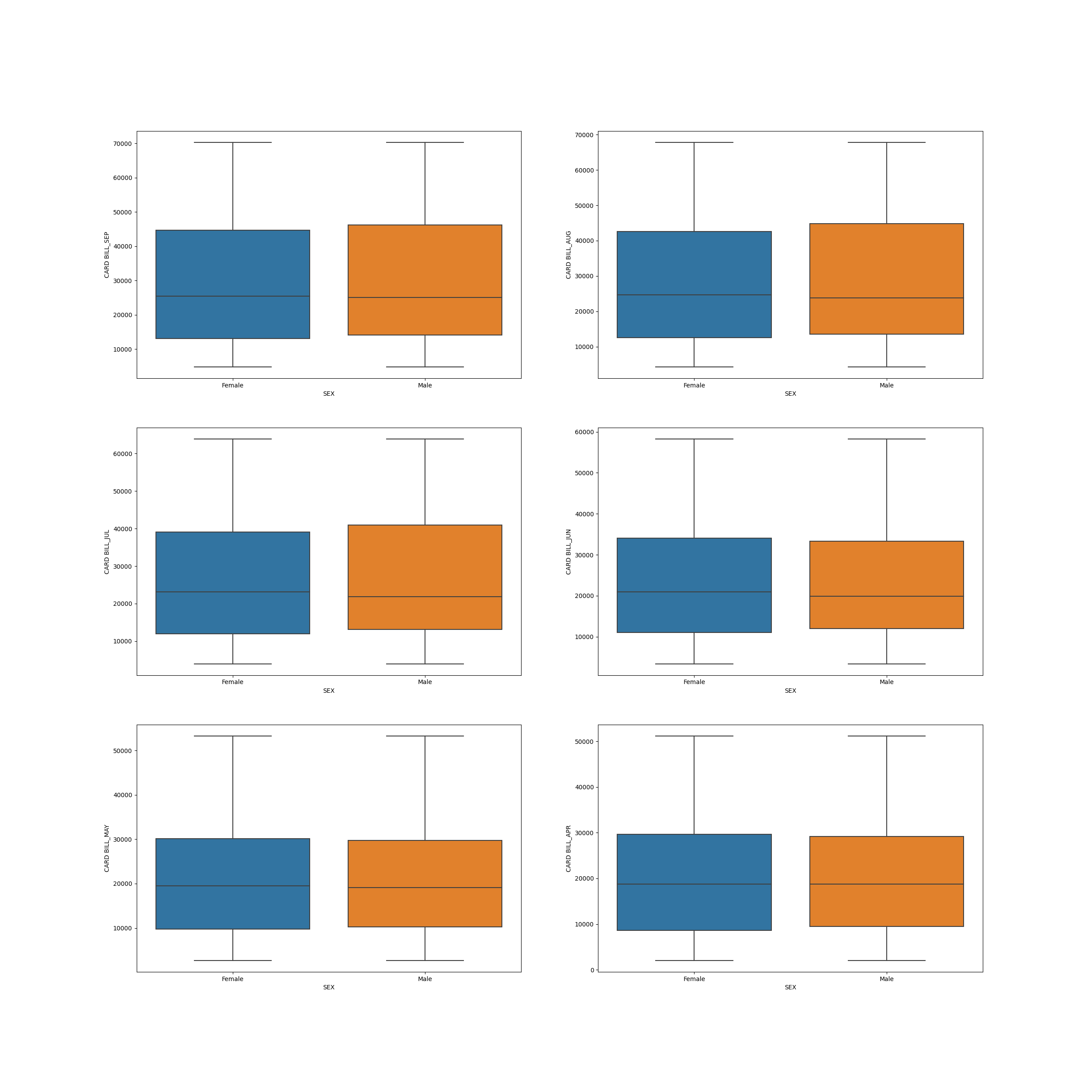
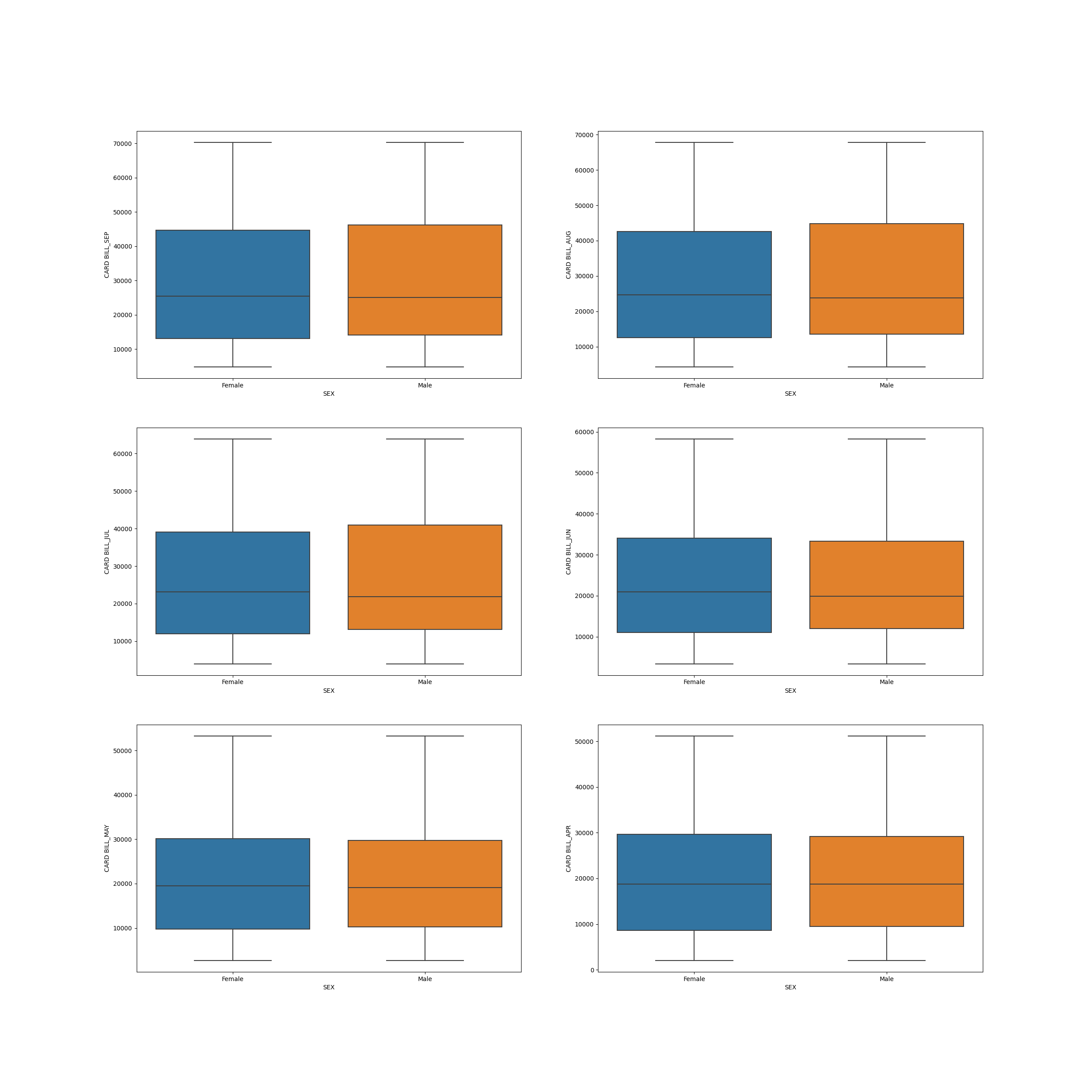


Fig1 = plt.figure(figsize = (25, 25))  
for i in range(len(RepayStatVariable)):  
 Fig1\_sub = Fig1.add\_subplot(3, 2, i + 1)  
 sns.countplot(x = dataframe['SEX'], hue = dataframe[RepayStatVariable[i]], hue\_order =  
 ['No Overdue', 'Short-term Overdue', '2 Months ~ 6 Months Overdue', 'Long-term Overdue'], data = dataframe)  
 pass  
plt.show()

성별에 따른 상환 현황의 차이 또한 거의 없었다. 9월을 제외하고 절대 다수의 신용카드 사용자가 연체없이 대금을 납부했다. 연체가 없는 고객을 제외하고는 2개월 ~ 6개월정도 연체한 고객이 많았으며 1개월 미만의 단기연체나 6개월 이상의 장기연체를 한 고객은 거

의 없었다. 9월은 다른 달보다 연체가 발생한 고객이 특히, 단기연체가 발생한 고객이 많 았지만 여전히 연체가 없는 고객이 절대다수였고 성별에 따라 차이를 보이지도 않았다.



CardBillVariable = ['CARD BILL\_SEP', 'CARD BILL\_AUG', 'CARD BILL\_JUL', 'CARD BILL\_JUN', 'CARD BILL\_MAY', 'CARD BILL\_APR']  
Fig2 = plt.figure(figsize = (25, 25))  
for i in range(len(CardBillVariable)):  
 Fig2\_sub = Fig2.add\_subplot(3, 2, i + 1)  
 Q1 = dataframe[CardBillVariable[i]].quantile(0.25)  
 Q3 = dataframe[CardBillVariable[i]].quantile(0.75)  
 IQR = Q3 - Q1  
 sns.boxplot(x=dataframe['SEX'], y=dataframe.loc[(dataframe[CardBillVariable[i]] > Q1 - 0 \* IQR)  
 & (dataframe[CardBillVariable[i]] < Q3 + 0 \* IQR)][CardBillVariable[i]], data=dataframe)  
 pass  
plt.show()

print(dataframe.groupby(['SEX']).median())

>>> CARD BILL\_SEP CARD BILL\_AUG CARD BILL\_JUL

SEX

Female 24110.0 23201.0 21865.0

Male 27284.0 26015.5 23610.0

CARD BILL\_JUN CARD BILL\_MAY CARD BILL\_APR

SEX

Female 20204.0 19225.0 18375.0

Male 20245.0 19374.0 19066.5

print(dataframe.groupby(['SEX']).std())

>>> CARD BILL\_SEP CARD BILL\_AUG CARD BILL\_JUL

SEX

Female 71740.557522 69743.446362 67307.687275

Male 78302.134347 75408.708089 72493.326428

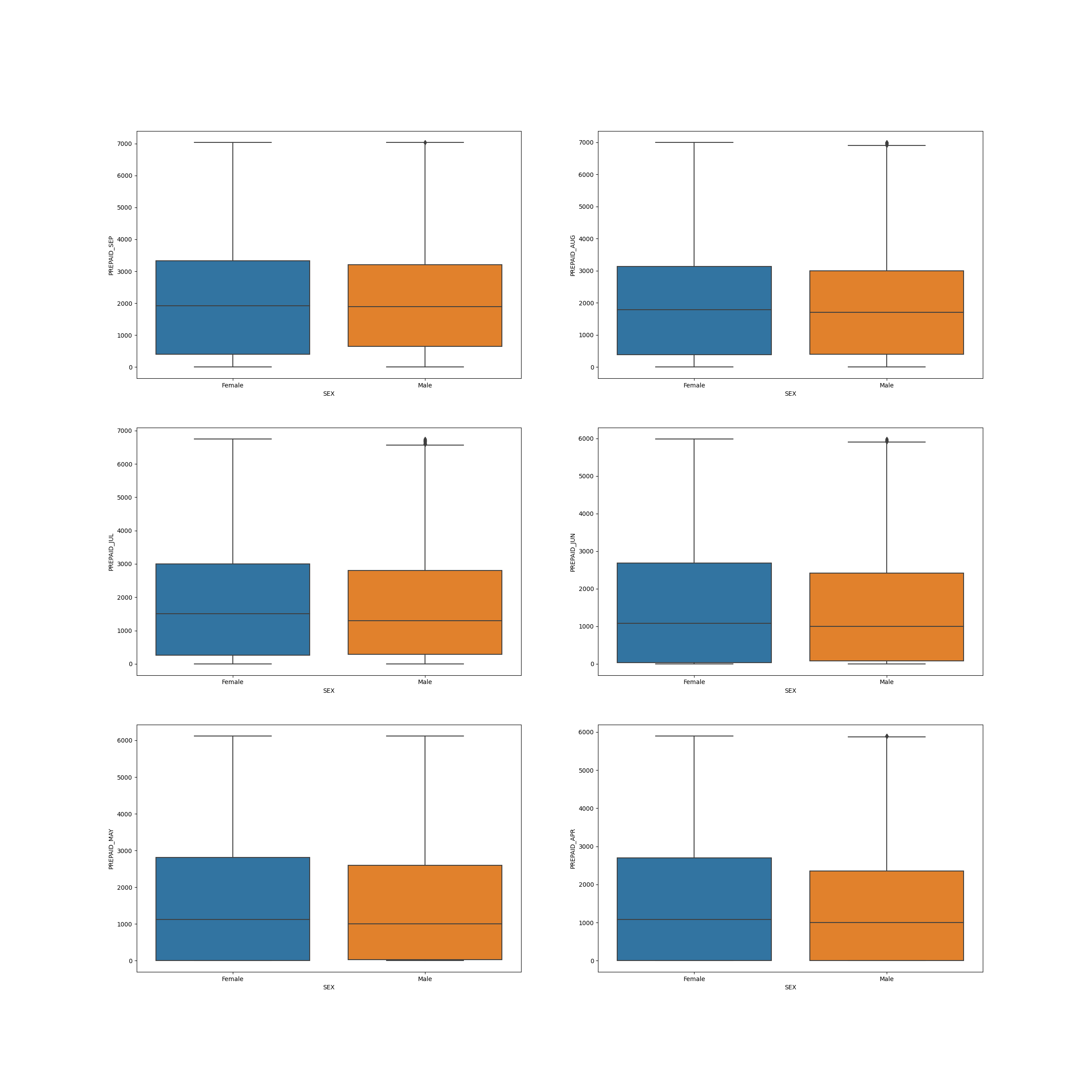
CARD BILL\_JUN CARD BILL\_MAY CARD BILL\_APR

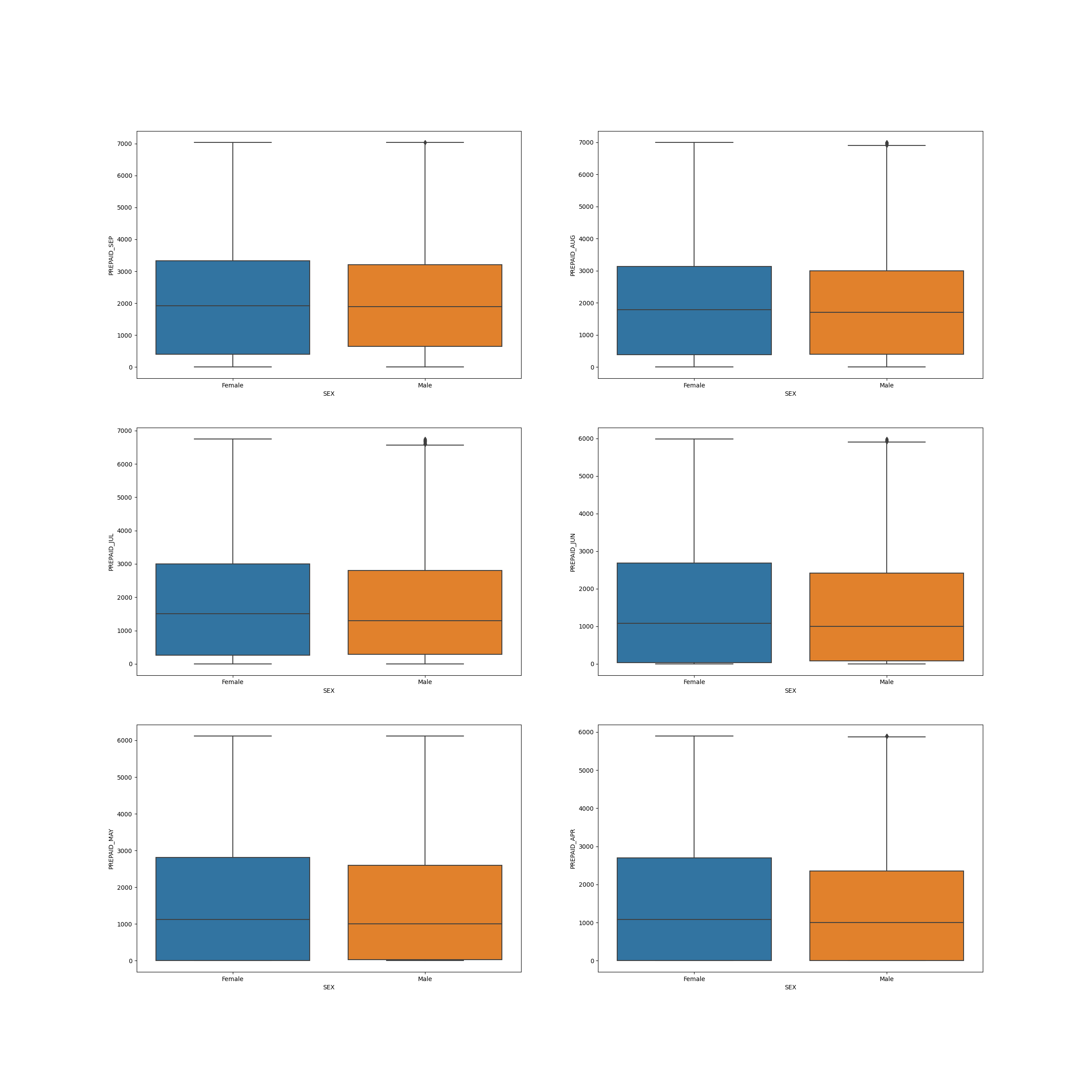
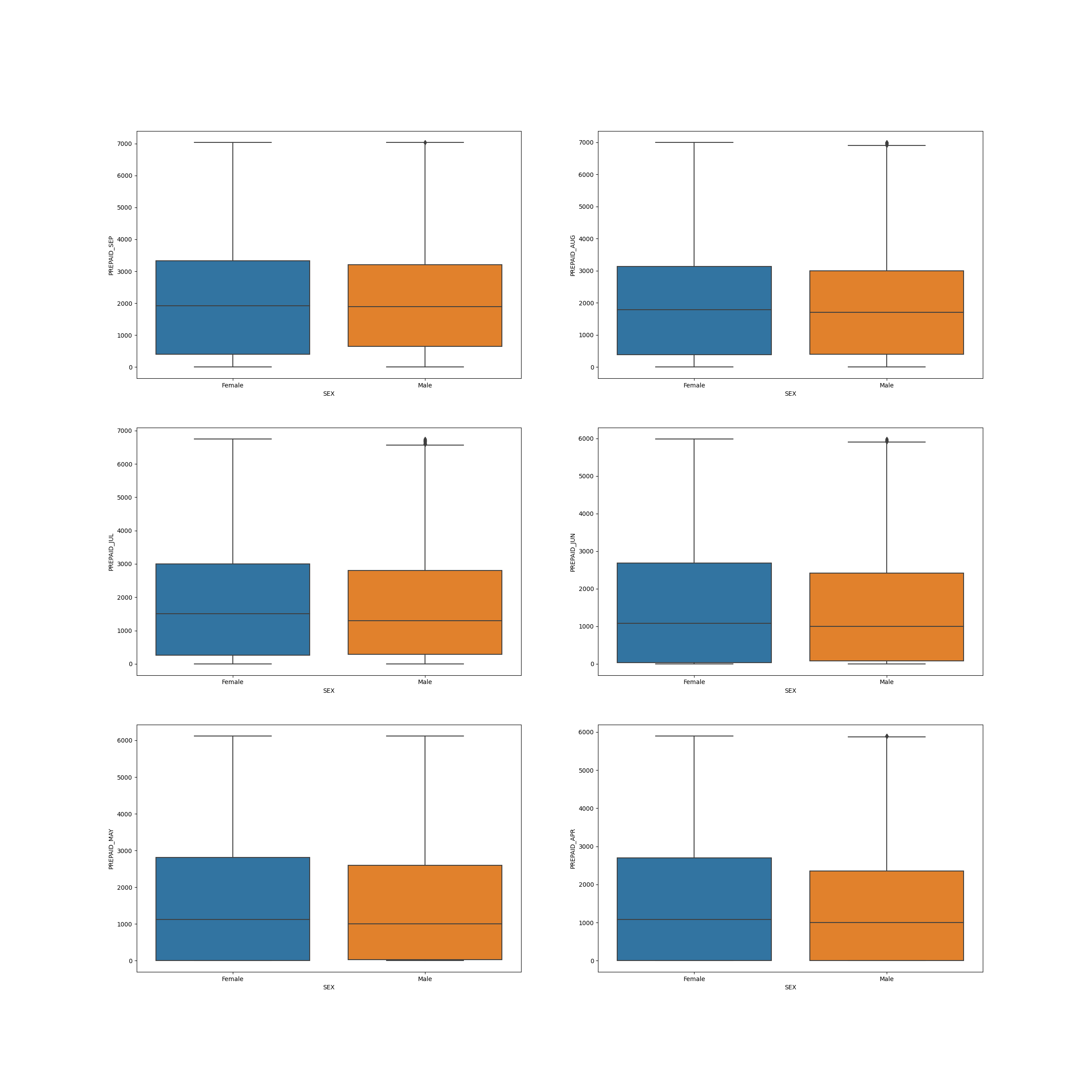
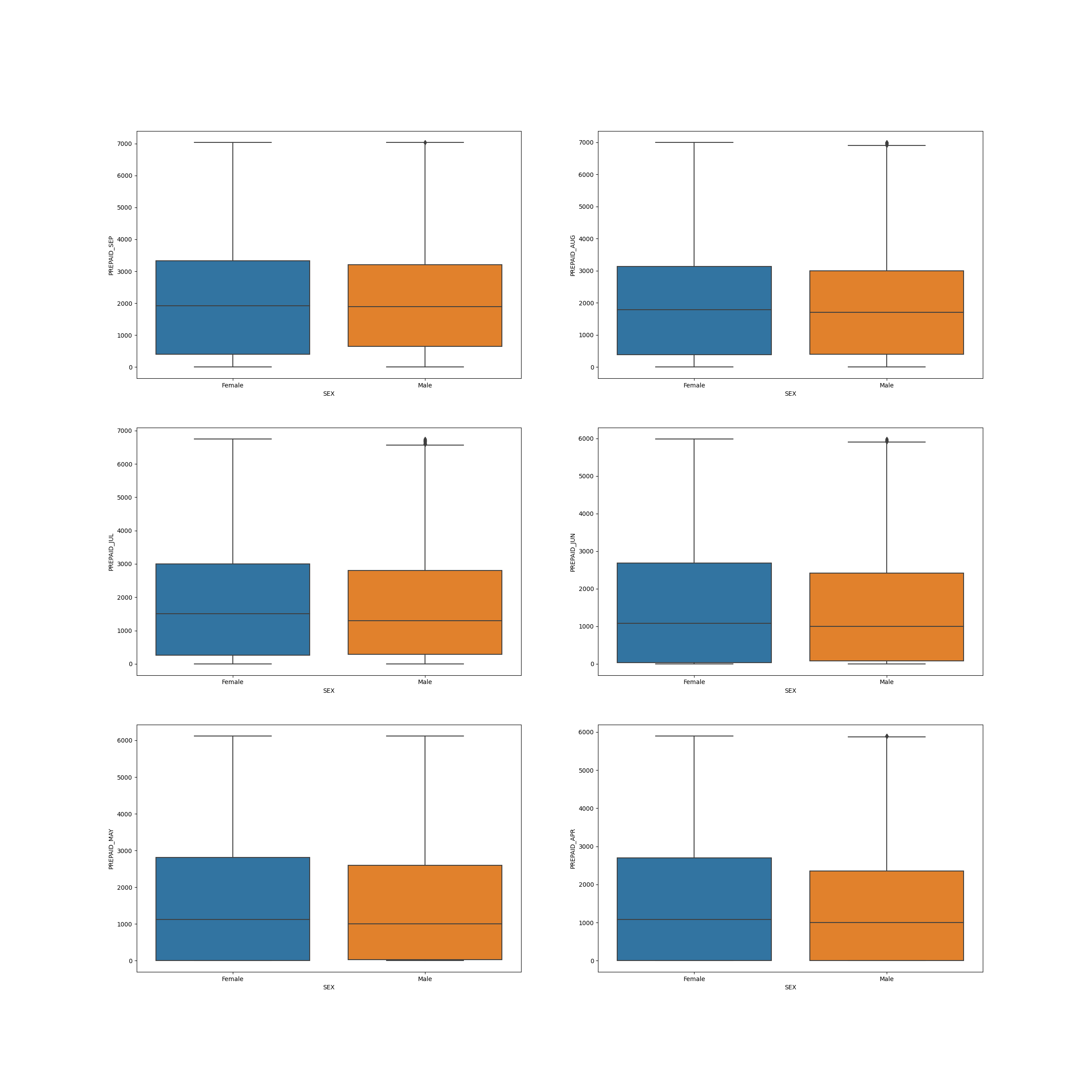
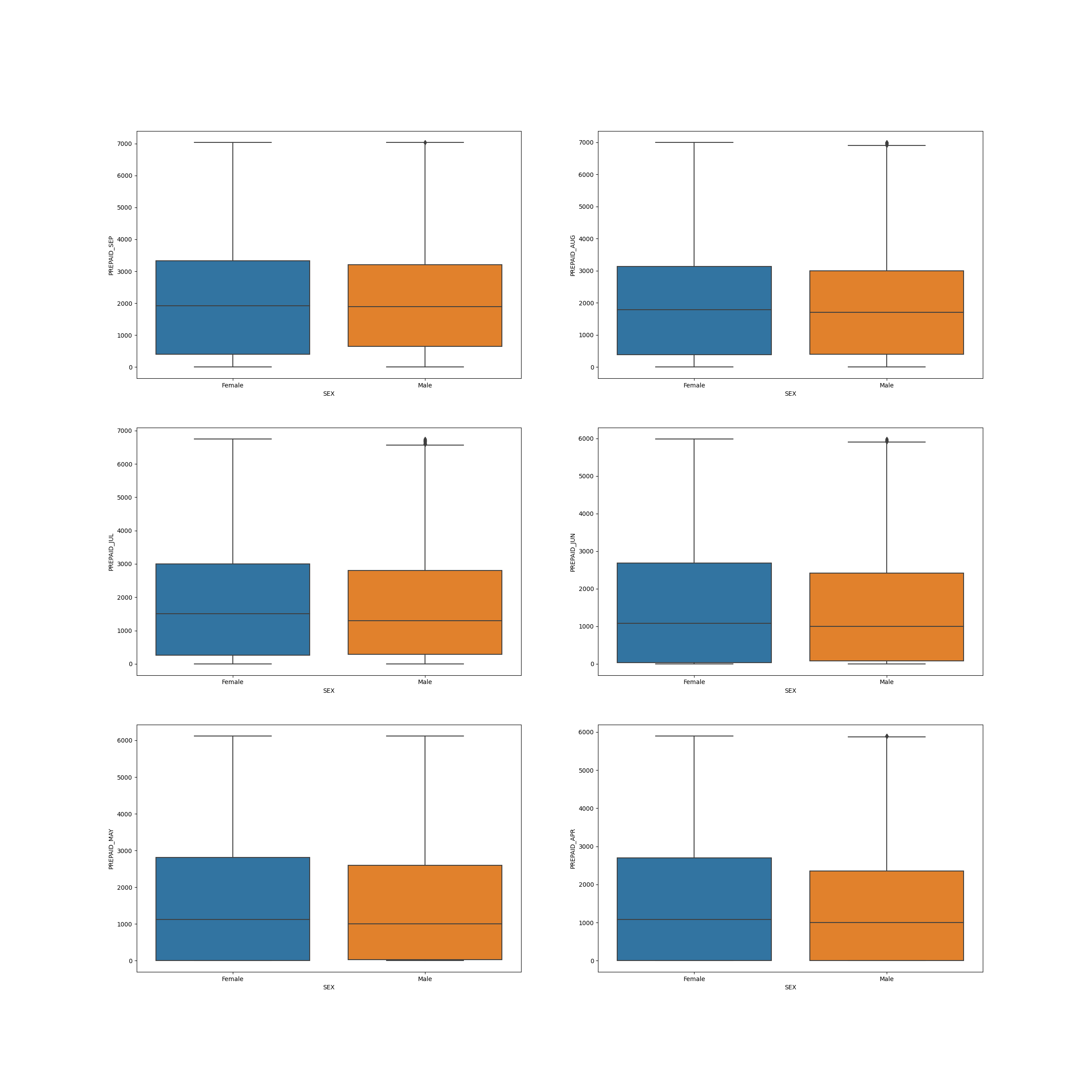
SEX

Female 63180.070518 59842.94055 58401.281674

Male 68547.740716 64617.94526 63521.645245

우선 아무 처리를 하지 않은 신용카드 대금 데이터는 이상치가 매우 많았고 1.5 IQR을 기준으로 처리를 하더라고 이상치가 많았으므로 백분위가 25 ~ 75인 관측값만 시각화했다. 이때 성별 에 따른 카드 대금 차이는 크지 않았다. 이상치를 포함한 표본을 기준으로 4월에서 9월로 갈수록 대금이 증가했으며 남성이 조금 더 많은 금액의 신용결제를 했다. 그러나 그 차이는 최대 약 10% 수준으로 큰 의미를 가진다고 보기 어렵다. 카드 대금의 표준편차는 남성이 더 높았다. 즉, 많이 결제하는 사람과 적게 결제하는 사람의 청구액 차이는 남성이 더 컸다.





PrepaidVariable = ['PREPAID\_SEP', 'PREPAID\_AUG', 'PREPAID\_JUL', 'PREPAID\_JUN', 'PREPAID\_MAY', 'PREPAID\_APR']  
Fig3 = plt.figure(figsize = (25, 25))  
for i in range(len(PrepaidVariable)):  
 Fig3\_sub = Fig3.add\_subplot(3, 2, i + 1)  
 Q1 = dataframe[PrepaidVariable[i]].quantile(0.25)  
 Q3 = dataframe[PrepaidVariable[i]].quantile(0.75)  
 IQR = Q3 - Q1  
 sns.boxplot(x=dataframe['SEX'], y=dataframe.loc[(dataframe[PrepaidVariable[i]] > Q1 - 0.5 \* IQR)  
 & (dataframe[PrepaidVariable[i]] < Q3 + 0.5 \* IQR)][PrepaidVariable[i]], data=dataframe)  
 pass  
plt.show()

print(dataframe.groupby(['SEX']).median())

>>> PREPAID\_SEP PREPAID\_AUG

SEX

Female 2230.0 2107.0

Male 2184.0 2021.0

PREPAID\_JUL PREPAID\_JUN PREPAID\_MAY PREPAID\_APR

SEX

Female 2000.0 1650.0 1700.0 1685.0

Male 1763.0 1500.0 1507.5 1431.5

print(dataframe.groupby(['SEX']).std())

>>> PREPAID\_SEP

SEX

Female 15598.474082

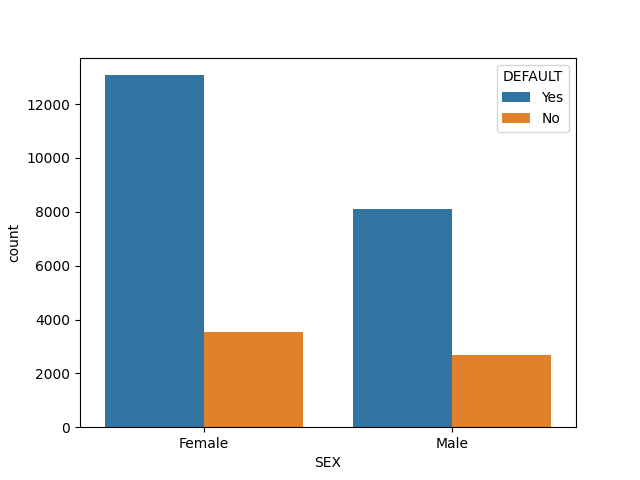
Male 15334.812381

PREPAID\_AUG PREPAID\_JUL PREPAID\_JUN PREPAID\_MAY PREPAID\_APR

SEX

Female 18871.692763 15374.810107 13541.053205 13968.128927 16398.998382

Male 17545.399612 17573.395222 15641.356428 15306.036719 18367.414843

선불 결제액의 경우에도 남녀의 차이가 크지 않았다. 이상치를 모두 포함한 표본을 기준으로 여성이 남성보다 상대적으로 선불결제를 더 많이 이용했다. 그 차이는 최대 약 17%로 이용 금액이 가장 적은 4월에 가장 컸다. 표준편차는 이상치를 제외한 기준으로는 여성이 항상 높았으나, 이상치를 포함하면 시기에 따라 탈랐지만 대부분 남성이 더 높았다.

sns.countplot(x = dataframe['SEX'], hue = dataframe['DEFAULT'], hue\_order = ['Yes', 'No'], data = dataframe)  
plt.show()

채무불이행 가능성에 있어서는 남녀는 다소 차이를 보였다. 남성은 10798명 중 2683명이 연체상태로 약 24.8%, 여성은 16599명 중 3524명이 연체상태로 약 21.2%를 보였다. 즉, 여 성보다 남성의 채무불이행, 연체 확률이 더 높았다.

정리하자면 평균적으로 부여받는 신용한도는 여성이 조금 더 높았다. 또한 사용패턴에 있어 서는 신용결제는 남성이 조금 더 많이 사용하는 경향이 있고, 선불결제는 여성이 조금 더 많 이 사용하는 경향이 있다. 연체 가능성에 있어서는 전반적인 패턴은 비슷하나 확률자체는 남 성이 유의미하게 높았다. 따라서 카드사라면 큰 차이는 아니지만 남성에게 상대적으로 적은 한도를 제시해 신용거래 결제금액을 줄여 연체에 빠질 가능성을 낮추는 것이 합리적이라고 볼 수도 있다.

그다음으로 교육수준에 따른 사용패턴과 채무불이행 가능성의 차이를 분석해보자